

# Interactive Close-Up Rendering for Detail+Overview Visualization of 3D Digital Terrain Models

Matthias Trapp  
Hasso Plattner Institute,  
Faculty of Digital Engineering,  
University of Potsdam, Germany  
matthias.trapp@hpi.de

Jürgen Döllner  
Hasso Plattner Institute,  
Faculty of Digital Engineering,  
University of Potsdam, Germany  
juergen.doellner@hpi.de

**Abstract**—This paper presents an interactive rendering technique for detail+overview visualization of 3D digital terrain models using interactive close-ups. A close-up is an alternative presentation of input data varying with respect to geometrical scale, mapping, appearance, as well as Level-of-Detail (LOD) and Level-of-Abstraction (LOA) used. The presented 3D close-up approach enables in-situ comparison of multiple Region-of-Interests (ROIs) simultaneously. We describe a GPU-based rendering technique for the image-synthesis of multiple close-ups in real-time.

**Keywords**—Terrain Visualization, Detail+Overview, Close-Up, Coordinated and Multiple Views

## I. INTRODUCTION

3D Digital Terrain Models (DTMs) [1] can be used as scenery to visualize surface related data, which is often represented by raster-images or surface textures. Common tasks for visualization and analysis include the comparison of different surface data sets *in-situ*, i.e., within the context of the same DTM or by using different geometric representations, e.g., LOD or LOA. Existing approaches that are capable of providing such functionality, are detail+overview visualization techniques [2], coordinated multiple views [3], and small multiples [4].

Further, detail-in-context techniques facilitate exploration of data spaces that are too large or have too much detail to fit in regular displays [5]. In addition thereto, *close-ups* are used in illustrations to provide detailed views on ROIs that are integrated into the rendering of the complete structure to retain their spatial context [6].

### A. Motivation

Extending the concept of 2D detail views to 3D has a number of advantages. The comparison of different surface data sets (e.g., height, slope, exposition) for the same ROI becomes easily possible. Besides that, a possibly important morphological aspect of a 3D DTM can be visualized from prioritized positions, so called “best views”. Further, automatic examination modes can display details without adding interaction tasks to the user – while maintaining the features of 2D detail+overview visualization as well.

That is, supporting multiple LOD or LOA [7] and enabling geometrical or geographical scaling by preserving context simultaneously. For example, Figure 1 depicts two mapping variants of the same 3D DTM within a single image using three different configurations of 3D close-ups.

In summary, such an approach enables a number of applications in the domain of geovisualization: *in-situ* comparison and exploration of different surface data (e.g., represented by different as surface textures or temporal variants) with only small amount of user interaction required and the combination of different LOA or representations of different information densities.

Further applications of this technique can be pre-flight briefings for airfields with difficult geographic surroundings, or for illustration purposes in modern, interactive teaching materials with 3D graphics, e.g., web-based encyclopedia.

### B. Related Work

Close-ups have a long history in map production. However, we focus on related work on detail+overview approaches in the domain of geovisualization. In their work Tominski *et al.* [8] present a general survey on the related topic of interactive lenses in visualization. Specifically, Karnik *et al.* present a route visualization technique based on detail lenses that addresses limitations of map displays with respect to driving directions [9]. The technique generates 2D route maps that shows the overview and detail views within a single, consistent visual frame.

Similar to our work, these detail views depicts ROI at a finer geospatial scale. Instead of using the same projection for detail and overview, “the undistort lens” [5] uses close-ups to combine details from separate ROIs for comparison, while retaining overview. Applications of interactive multimodal 2D and 3D close-ups to medical reporting based on volume data sets are presented in [6]. Similar to our approach, it supports different visualization styles and relies on a Graphics Processing Unit (GPU)-based rendering algorithm for multiple closeups at interactive frame rates.

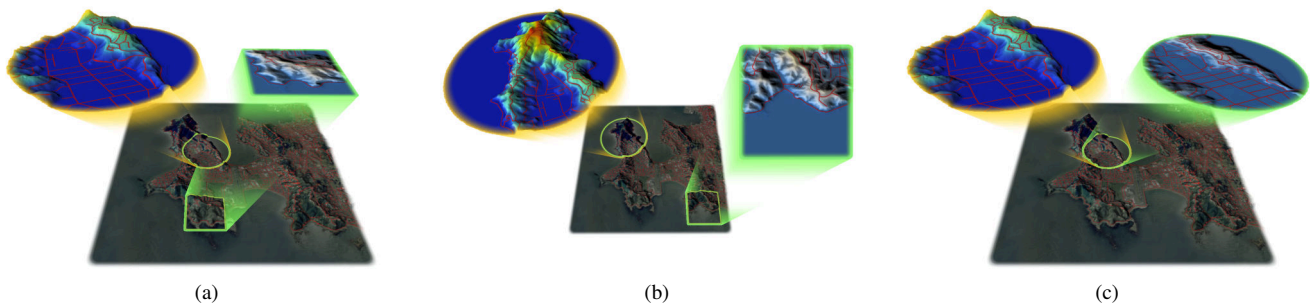


Figure 1: Exemplary results using two 3D close-ups with different data mappings and variations of orientation and projection.

### C. Problem Statement and Contributions

Enabling interactive 3D close-ups for DTM raises the conceptual and technical challenges of how to (1) *model* and *parametrize*, (2) *interact* with, and (3) *render* such visualization in real-time.

While detail views are a well studied concept in visualization, their application within geovisualization of 3D DTMs is, to the best knowledge of the authors, unexplored. With respect to the challenges above, this paper presents a visualization concept and interactive rendering technique for 3D close-ups that facilitates an integrated detail+overview approach for 3D DTMs. It enables the usage of different perspectives for the overview and individual detail views.

The remainder of this work is structured as follows: Section II introduces the concept of 3D interactive close-ups. Section III present interaction techniques specific for the effective usage of close-ups. Section IV describes implementation details of our interactive rendering technique. Finally, Section V concludes this work and present ideas for future research.

## II. PRINCIPLES OF CLOSE-UP VISUALIZATIONS

This section briefly describes the basic concept, the design principles, and the parametrization of 3D close-ups. Figure 2 shows an overview of the presented approach and the

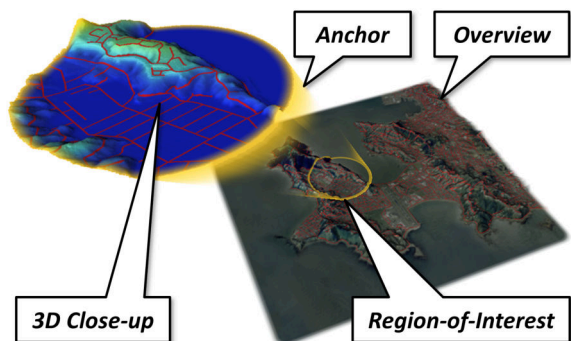


Figure 2: Visual components of a 3D close-up visualization.

terminology used through this paper. It basically comprises the following visual components: (1) a single *overview* [10] represents a visualization context of the 3D DTM using a respective data mapping (aerial image); (2) a *Region-of-Interest* of potentially arbitrary shape (here circle) indicates the extend of the (3) 3D *close-up* on the overview; finally, (4) an *anchor* that connects the close-up with the ROI.

### A. Design Principles

The application of overview+detail or detail-in-context visualization principles have a long tradition in manufacturing maps and information graphics. For example, Figure 3 shows close-up illustrations that use different stylizations and projections. Based on these artifacts, the following design principles could be identified for close-ups (besides positioning and layout):

1) *Information Density (P1)*: The information density of close-ups is higher than of the overview. Often, close-ups show different mappings or presentation forms of the data to visualize. This can include LOD as well as LOA.

2) *Close-up Referencing (P2)*: An anchor is usually depicted using outlines and color variations. They typically do not exhibit lighting or shading.

3) *Orientation and Projection (P3)*: Besides exceptions (e.g., [5]), close-ups often use the same orientation and projection as the overview does. However, orientations can be chosen to facilitate “best views” of ROIs.

4) *Number of Close-ups (P4)*: In general, close-ups are sparsely used since a high number would often increased occlusion of the overview.

Based on these principles, the remainder of this section describes the parametrization for the particular visualization components.

### B. Close-up Parameters

A 3D close-up can basically parameterized by the following components: (1) a *geometric transformation* for positioning and orientation, (2) a *ROI mask*, and (3) a *color*. The latter facilitates association of close-up and ROI (cf. P2). The geometric transformation required for a particular close-up and

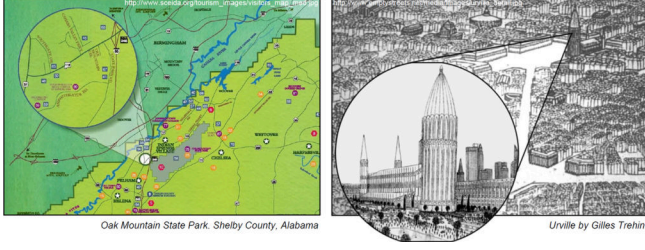


Figure 3: Examples of hand-crafted illustrations using 2D and 3D close-ups for detail+overview visualization.

its respective anchor can be performed on a per vertex basis, e.g., by using vertex or geometry shader functionality [11]. For a given vertex of the overview representation  $V_O$  in object-space coordinates, the transformed close-up vertex  $V_C$  is basically computed using a transformation matrix  $\mathbf{C}$ , yielding:  $V_C = \mathbf{MVP} \cdot \mathbf{C} \cdot V_O$  with  $\mathbf{C} = m \cdot \mathbf{M} + (1 - m) \cdot \mathbf{I}$  being the linear interpolation of the close-up transformation matrix  $\mathbf{M}$  and the identity matrix  $\mathbf{I}$ , controlled by a mask factor  $m$ . This factor is obtained for each input vertex by dynamically mapping the mask texture according using projective texturing according to [12] and controlled using a ROI origin  $O = (x, y)$  as well as an associated scaling factor. The  $\mathbf{MVP}$  matrix represents the model-view-projection transformation used for the remaining vertex transform prior to rasterization [13]. The close-up transformation matrix is defined by:

$$\mathbf{M} = \mathbf{R}(\alpha, \beta) \cdot \mathbf{S}(s_x, s_y, s_z) \cdot \mathbf{T}(\vec{p})$$

which is controlled by the following parameters:

$\mathbf{R}(\alpha, \beta)$ : This rotation matrix defines the close-up orientation (cf. P3). It is composed of two rotations, one around the  $x$ -axis ( $\alpha$ ) and one around the  $y$ -axis ( $\beta$ ), respectively.

$\mathbf{S}(s_x, s_y, s_z)$ : This scaling transformation defines the extend of the close-up ( $s_x$  and  $s_z$ ) as well as the depicted DTM height, i.e., setting  $s_y = 0$  yield a flat terrain in the close-up (Figure 1).

$\mathbf{T}(\vec{p})$ : This translation matrix encodes the displacement of the close-up in relation to the overview. Note that the  $y$ -coordinate should be of positive range.

### C. Close-up Orientation and Projection

During interaction with the overview, the positioning and orientation of the close-ups can be configured individually. The parametrization supports mainly two different variants: (1) *overview-dependent* close-ups that use the same orientation and projection configuration as the virtual camera configuration of the overview (Figure 1a); and (2) *overview-independent* close-ups that use a fixed orientation independent of the overview. As special case, a billboard transformation in combination with an orthographic projection can be applied (Figure 1b).

### D. Anchor Parameters

The close-up anchor associates the ROI and the close-up using a fence-like depiction. To facilitate referencing of a close-up to its ROI on the overview, the anchor is colored using the respective close-up color. To reduce occlusions of overview, the anchor transparency can be adjusted accordingly based on the height differences between the overview and the close-up. For it, the respective distances between  $V_O$  and  $V_C$  are normalized to the range  $[0, 1]$  and subsequently used to sample a 1D texture representing transparency values used in the final compositing pass. For example, Figure 4 shows the impact of different textures to the amount of overview occlusion and referencing effectiveness.

## III. INTERACTION TECHNIQUES FOR 3D CLOSE-UPS

This section briefly describes the potential of interaction techniques for the creation and manipulation of close-up visualizations.

### A. Region-of-Interest Definition

To effectively use the presented visualization approach, a user should be able to easily create and manipulate the ROI with respect to its shape, position, and size. To facilitate this, there are the following three possibilities to define a ROI:

1) *Shape Library*: A user can select and place a predefined ROI mask offered by the system's shape library. These 2D texture represent the shape using gray scale values.

2) *Shape Painting*: To allow for custom, user controlled shapes, the respective texture mask can be created by using direct-manipulation metaphors, e.g., a painting metaphor [14].

3) *Shape Generation*: The shape of a close-up can be generated or derived based on selected features of the 3D DTM (e.g., geometry or texture features) [15].

### B. Close-up State and Animation

To reduce visual clutter and occlusion when using multiple 3D close-ups, it is feasible to distinguish between *active* and *inactive* states.

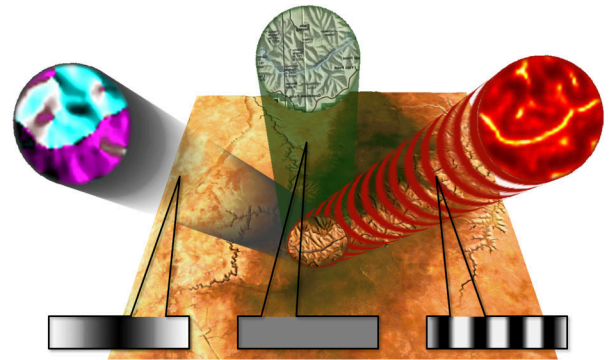


Figure 4: Multiple close-ups using different anchor stylizations.

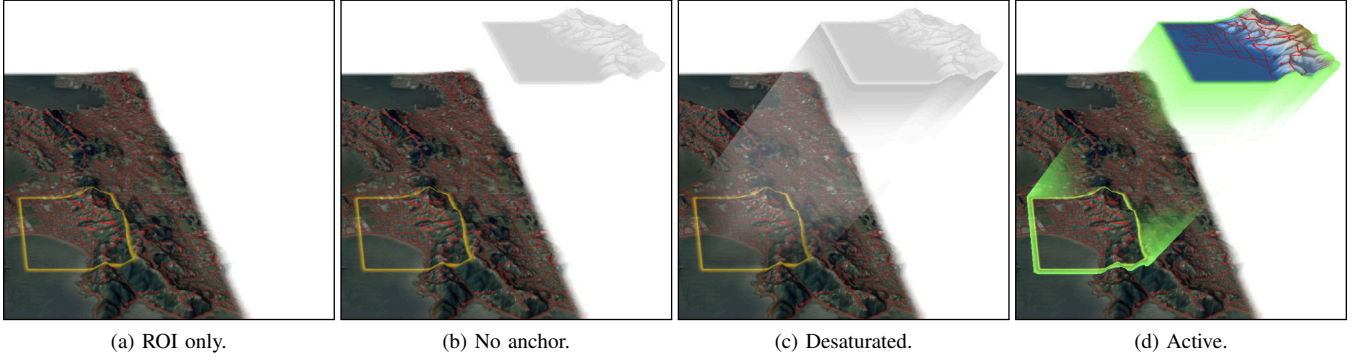


Figure 5: Depiction variants for inactive close-ups.

Figure 5(a) to (c) shows three variants for depicting inactive close-ups; differing with respect to the anchor visibility and coloring. To toggle between active and inactive states, a user can perform one of three actions while hovering over or clicking: (1) the ROI, and (2) the close-up, or using an additional mode key, respectively.

To support spatial exploration, the presented approach allows for automatic animation of close-ups. While a user controls the close-up position interactively, the orientation transformation can be rotated automatically to show its content from different point-of-views. The animation speed and repetition can be set for each close-up separately.

#### IV. INTERACTIVE RENDERING OF 3D CLOSE-UPS

This section briefly describes details for implementing an interactive rendering technique (using OpenGL [16]) that is capable of performing image synthesis of the proposed concept in real-time.

##### A. Rendering Pipeline Variants

Due to their potentially high geometric complexity (i.e., graphics primitives), the rendering of DTMs can be a costly process (in terms of run-time). Therefore, the implementation of a rendering technique should require only a minimum number of forward passes [13] for multiple close-ups (cf. P4). Since the close-up parameters are defined with respect to the context, close-up images are required to be re-rendered on a per-frame basis. There are basically two options for the design of a rendering pipeline depending on the geometry of overview and close-ups (cf. P1):

1) *Single-pass Forward Rendering (SPFR)*: Using a *single-pass rendering* approach is favorable if the same geometry for overview and close-up is used. Variances in data mapping (using texturing, lighting, and stylization) can be performed using image-based techniques.

2) *Multi-pass Forward Rendering (MPFR)*: If different LOD or LOA are present for the overview and the particular close-ups, a traditional *multi-pass rendering* approach is

required comprising one pass for the close-up and one for the anchor respectively.

Figure 6 shows an overview comprising operations, data sources, as well as control and data flow. For the SPFR approach, it basically comprises three principle stages: (1) *geometry amplification and close-up transformation*, (2) *layered rendering*, and finally (3) *image-based compositing*.

During geometry amplification, the input primitives are duplicated according to the number of close-ups to render, the respective geometric transformations are applied on a per vertex basis, and respective layer ids required for layered rendering are assigned to each primitive. This stage can be implemented using a geometry shader and thus is agnostic to terrain data that is streamed out-of-core. Subsequently, layered rendering is performed by rasterizing the amplified primitives to the corresponding image-based intermediate representations [17], organized as a single 2D texture array.

##### B. Data Representation and Compositing

To facilitate efficient real-time rendering, a major challenge concerns data structures that are suitable for GPU-based implementations:

1) *Configuration Data*: To support SPFR, all close-up parameters are represented and encoded using a single uniform buffer object, which enables structured access by shader programs and can be partially updated.

2) *Surface Data*: For representation of multiple surface textures for a DTM and to support its efficient rendering based on texturing, the individual data sets presented by 2D textures are organized using a 2D texture array that support access by indexing.

3) *Mapping Data*: Besides the computation of local lighting models, a data mapping representation mainly comprises texture mapping. This includes additional data such as color maps. For it, particular shader subroutines for each close-up are used.

To synthesize the final visualization output, a *compositing rendering pass* is performed that combines intermediate overview and close-up renderings using blending operations.



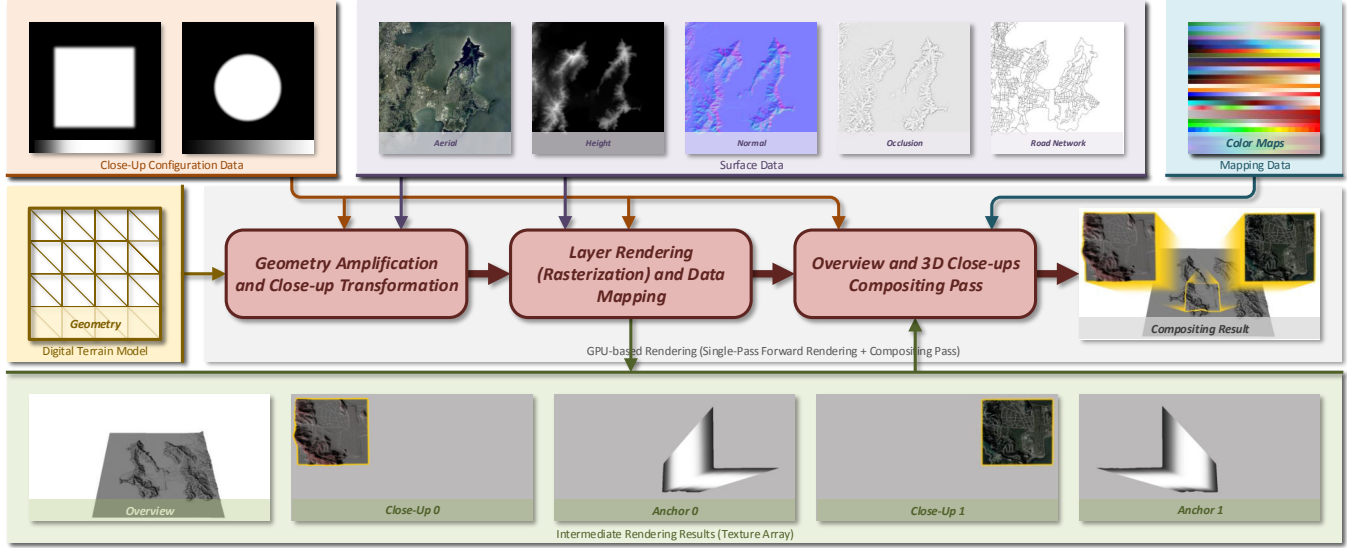


Figure 6: Schematic rendering pipeline comprising control (red) and data flow (other colors) as well as rendering components.

By texturing a screen-aligned quad covering the viewport, the 2D texture array storing the results from the previous forward pass(es) are sampled and blended in the order of generation. This step is performed using fragment shader for increased flexibility during compositing.

### C. Performance and Limitations

We tested the rendering performance of a prototypical SPFR implementation using a 3D DTM represented by a regular grid of 5 659 666 vertices and 11 290 784 triangle primitives (indexed). The performance test was conducted using a NVIDIA GeForce GTX 970 GPU with 4096 MB VRAM on a Intel Xeon CPU with 2.8 GHz and 12 GB RAM rendering at a viewport resolution of  $1280 \times 720$  pixels. The run-time performance mainly depends on the geometric complexity of the 3D scene and decreases linearly with respect to the number of close-ups used. Table I shows the obtained measurements in FPS, averaged over 500 frames.

The presented approach has conceptual and technical limitations. First, the number of simultaneously depicted close-ups is limited due to the trade-off between available screen-space for overview and close-ups. Further, the proposed SPFR implementation requires a sufficient vertex density to achieve high-quality rendering of close-up anchors. The

close-up are manually configured and layouts, thus possible close-up overlaps are not handled automatically. The linear performance is due to the reason that the complete terrain geometry is rendered for each close-up, thus putting unnecessary stress on the vertex and/or geometry shader processing. However, this can be optimized by using culling techniques.

### V. CONCLUSIONS AND FUTURE WORK

This paper presents an interactive visualization technique featuring 3D close-ups for in-situ comparison of surface data of 3D DTM. Based on real-world illustrations, design principles are derived and respective visualization parameters are defined. The presented concept was prototypical implemented by a GPU-based rendering technique, which specifics are briefly described.

Based on the current limitations, there are various ways for future research. First, a layout component that automatically computes close-ups layouts and their configuration in a view-based approach can be developed. This component can also control more elaborated animation functions, e.g., orbiting cameras. For rendering, one can exploit tessellation capability of GPUs to ensure sufficient vertex density.

### ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable feedback and suggestions to improve the paper. This work was funded by the Federal Ministry of Education and Research (BMBF), Germany, within the InnoProfile Transfer research group “4DnDVis”

Table I: Results of run-time performance evaluation.

Number of 3D Close-ups	Frames-per-Second (FPS)
0	517.9
1	226.5
2	142.1
3	104.8
4	81.2

## REFERENCES

- [1] C. Hirt, “Digital Terrain Models,” *Encyclopedia of Geodesy*, 06 2016.
- [2] A. Cockburn, A. Karlson, and B. B. Bederson, “A review of overview+detail, zooming, and focus+context interfaces,” *ACM Comput. Surv.*, vol. 41, no. 1, pp. 2:1–2:31, Jan. 2009.
- [3] J. C. Roberts, *Coordinated Multiple Views for Exploratory GeoVisualization*. John Wiley & Sons, Ltd, 2008, ch. 3, pp. 25–48.
- [4] S. v. d. Elzen and J. J. v. Wijk, “Small Multiples, Large Singles: A New Approach for Visual Data Exploration,” *Computer Graphics Forum*, 2013.
- [5] J. Brosz, S. Carpendale, and M. A. Nacenta, “The Undistort Lens,” *Computer Graphics Forum*, vol. 30, no. 3, 2011.
- [6] T. Ropinski, I. Viola, M. Biermann, H. Hauser, and K. Hinrichs, “Multimodal Visualization with Interactive Closeups,” in *Theory and Practice of Computer Graphics*, W. Tang and J. Collomosse, Eds. The Eurographics Association, 2009.
- [7] A. Semmo, M. Trapp, J. E. Kyprianidis, and J. Döllner, “Interactive Visualization of Generalized Virtual 3D City Models using Level-of-Abstraction Transitions,” *Computer Graphics Forum*, vol. 31, no. 3, pp. 885–894, 2012, proceedings EuroVis 2012.
- [8] C. Tominski, S. Gladisch, U. Kister, R. Dachsel, and H. Schumann, “A Survey on Interactive Lenses in Visualization,” in *EuroVis - STARs*, R. Borgo, R. Maciejewski, and I. Viola, Eds. The Eurographics Association, 2014.
- [9] P. Karnik, D. Cline, S. Jeschke, A. Razdan, and P. Wonka, “Route Visualization using Detail Lenses,” *IEEE Transactions on Visualization and Computer Graphics*, vol. 16, no. 2, pp. 235–247, 2009.
- [10] K. Hornbæk and M. Hertzum, “The Notion of Overview in Information Visualization,” *International Journal of Human-Computer Studies*, vol. 69, no. 7-8, pp. 509–525, 2011.
- [11] M. Segal and K. Akeley, *The OpenGL Graphics System: A Specification (Version 4.6 (Core Profile) - May 14, 2018)*, The Khronos Group Inc., May 2018.
- [12] M. Trapp and J. Döllner, “Dynamic Mapping of Raster-Data for 3D Geovirtual Environments,” in *13th International Conference on Information Visualisation, IV 2009, 15-17 July 2009, Barcelona, Spain*, 2009, pp. 387–392.
- [13] T. Akenine-Möller, E. Haines, N. Hoffman, A. Pesce, M. Iwanicki, and S. Hillaire, *Real-Time Rendering 4th Edition*. Boca Raton, FL, USA: A K Peters/CRC Press, 2018.
- [14] M. Trapp, C. Beesk, S. Pasewaldt, and D. Jürgen, “Interactive Rendering Techniques for Highlighting in 3D Geovirtual Environments,” in *Proceedings of the 5th 3D GeoInfo Conference*, ser. Lecture Notes in Geoinformation & Cartography. Springer, 11 2010.
- [15] M. Trapp, T. Glander, H. Buchholz, and J. Döllner, “3D Generalization Lenses for Interactive Focus + Context Visualization of Virtual City Models,” in *Information Visualisation, 2008. IV '08. 12th International Conference*, July 2008, pp. 356–361.
- [16] M. Segal and K. Akeley, *The OpenGL Graphics System: A Specification Version 4.5*, 4th ed., Silicon Graphics Inc., June 2017.
- [17] T. Saito and T. Takahashi, “Comprehensible Rendering of 3-D Shapes,” *SIGGRAPH Comput. Graph.*, vol. 24, no. 4, pp. 197–206, Sep. 1990.